4

USAISEC

US Army Information Systems Engineering Command
Fort Huachuca, AZ 85613-5300

U.S. ARMY INSTITUTE FOR RESEARCH
IN MANAGEMENT INFORMATION,
COMMUNICATIONS, AND COMPUTER SCIENCES
(AIRMICS)

SOFTWARE TOOLS FOR SOFTWARE MAINTENANCE

(ASQBG-I-89-001)

October, 1988

AIRMICS
115 O'Keefe Building
Georgia Institute of Technology
Atlanta, GA 30332-0800

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188
Exp. Date: Jun 30, 1986

1a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b. RESTRICTIVE MARKINGS
NONE

2a. SECURITY CLASSIFICATION AUTHORITY
N/A

2b. DECLASSIFICATION / DOWNGRADING SCHEDULE
N/A

3. DISTRIBUTION / AVAILABILITY OF REPORT
N/A

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
ASQBG-I-89-001

5. MONITORING ORGANIZATION REPORT NUMBER(S)
N/A

6a. NAME OF PERFORMING ORGANIZATION
AIRMICS

6b. OFFICE SYMBOL (if applicable)
ASQBG - I

7a. NAME OF MONITORING ORGANIZATION
N/A

6c. ADDRESS (City, State, and ZIP Code)
115 O'Keefe Bldg.,
Georgia Institute of Technology
Atlanta, GA 30332-0800

7b. ADDRESS (City, State, and Zip Code)
N/A

8a. NAME OF FUNDING/SPONSORING ORGANIZATION
AIRMICS

8b. OFFICE SYMBOL (if applicable)
ASQBG - I

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
N/A

8c. ADDRESS (City, State, and ZIP Code)
115 O'Keefe Bldg.,
Georgia Institute of Technology
Atlanta, GA 30332-0800

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO
62783A	DY10	02-03-01	

11. TITLE (Include Security Classification)
Software Tools for Software Maintenance (UNCLASSIFIED)

12. PERSONAL AUTHOR(S)
John Leopard

13a. TYPE OF REPORT
Final

13b. TIME COVERED
FROM ______ TO ______

14. DATE OF REPORT (Year, Month, Day)
October 1988

15. PAGE COUNT
52

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Software, Maintenance, Software Tools

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Software tools for software maintenance are presented along with tables by tool function, selecting those which are most readily adaptable to the U.S. Army Information Systems Engineering Command environment.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED / UNLIMITED □ SAME AS RPT. □ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL
John Leopard

22b. TELEPHONE (Include Area Code)
(404) 894-3110

22c OFFICE SYMBOL
ASQBG - I

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

This work was done as part of an in-house research project within the United States Army Institute for Research in Management Information, Communications, and Computer Sciences (AIRMICS), the RDTE organization of the United States Army Information Systems Engineering Command (USAISEC). This report is not to be construed as an official Army position, unless so designated by other authorized documents. Material included herein is approved for public release, distribution unlimited. Not protected by copyright laws.

THIS REPORT HAS BEEN REVIEWED AND IS APPROVED.

s/
Glenn E. Racine, Chief
Computer & Information
Systems Division

s/
John R. Mitchell
Director
AIRMICS

TABLE OF CONTENTS

TABLES

SOFTWARE TOOLS FOR SOFTWARE MAINTENANCE

Software maintenance continues to be among the most critical issues facing all software development organizations. Developers are facing increasing demands for new software applications while at the same time having to devote sizable resources to maintaining existing applications. It seems that software development organizations will reach steady-state when all available resources are devoted to software maintenance. Increasing the productivity of software developers and increasing the quality of systems produced is certainly a high priority in development organizations; however, software maintenance productivity seems to be at least as important but not as eagerly addressed.

Software maintenance has been reported as containing 70-80% of the total software life cycle costs [FC1]. In the federal government software maintenance is reported to cost an estimated $5 billion [FC2]. In a summary of 75 major companies, 53% of the total software related activities is in maintenance. In a survey conducted by IBM, over 80% of data processing resources are allocated to maintenance. Even a small increase in productivity in software maintenance will have a dramatic impact on overall data processing costs.

There has been much written about software development workbenches, computer aided software engineering, etc. but most of these fail to address the unique needs of maintenance programmers.

The most important impediment to maintaining software is understanding what the existing software does. This is difficult because the maintenance programmer is usually forced to use the code itself as the best documentation of the program. The requirements documents, design documents, comments in the code, etc. are often not kept up-to-date and programmers cannot use this documentation with confidence.

As part of an effort by AIRMICS to explore maintenance problem areas in order to specify an integrated set of tools that should be included in a Maintenance Work Station, a survey of software maintenance tools was conducted by AIRMICS in conjunction with a similar effort by the Purdue/University of Florida Software Engineering Research Center.

There are a considerable number of software maintenance support tools available, however, there are relatively few (maybe none?) sets of integrated maintenance tools.

The tools reported on in this survey are categorized by operating environment and by 13 different maintenance support functions, by 10 different languages supported, and by the source of information.

The tools in both the SERC and AIRMICS surveys, Appendices A and B, respectively, are listed in alphabetical order by product name.

Several software tools can be employed to make systems more maintainable. These tools provide automated documentation in a standard format, revers engineering tools to work backwards toward the original requirements, code reformatters to make code easier to read and logic easier to follow, code restructurers to optimally reorganize source code making it more structured and modular, data-name standardizers to provide for consistent data references in the source code, and many tools for managers to use in the control of maintenance activities.

A possible grouping of such tools would be:

1. Change Justification (as opposed to redesign):

   a. cost-benefit analysis tool

   b. cost estimating tool

   c. requirements specification tool

2. Management tools:

   a. automated progress and status reporting tool

   b. configuration management tool

   c. project budgeting, planning, scheduling, and control tool

3. Aids to understanding existing system:

   a. data tracking and cross referencing tool

   b. documentation tool

   c. source code analyzer

4. Tools for testing changes:

   a. editor that predicts impact of changes by working with data tracking tool

b. test file generator

c. executive or job control language generator (emulating production)

d. test coverage monitor tool

e. source and file compare tool

5. Tools for use in preventive maintenance

   a. data name standardization tool

   b. code restructuring tool

   c. code reformatter tool

   d. configuration management tool to maintain sets of tests for next change

   e. metrics analyzer tool to measure effectiveness of maintenance group

6. Translation tool for use in preparing systems to run in more than one environment.

To consolidate/provide a concise view of these tools appropriate to ISEC we have provided below a listing by function supported of these tools that are useful with the COBOL and IBM environments. Additional information about the tools listed below can be found in the appendices. Tools are included for IBM Job Control Language (JCL) and those selected for the GSA Programmers Workbench are noted "(PWB)."

Code Analyzer Tools

Environment	Language	Product Name	App
Custom Fitted	None Spec	ACT	B
None Specified	High Level	Battlemap	B
IBM Main MVS,DOS and OS	VS Cobol II	COBOL Structuring Facility	A
IBM Main MVSwTSO and VMwCMS	Cobol	Fastbol	A
IBM OS/VS	Cobol	Inspector	B
IBM Main DOSwVSE MVS and OS	Cobol	LogiChain	A
None Specified	Cobol	MAP	B
IBM Main OS, MVS	Cobol	Pathvu	A
IBM Main MVS, OSwTSO	Cobol	PM/SS	A
None Specified	Cobol/JCL	PSA/PSL	B
IBM Main	Cobol	RETROFIT	B
None Specified	Cobol	Reverse Engineering	A
IBM Main	Cobol	SAGE Maintenance Programming System	A
IBM Main DOS,MVS OS and VM	Cobol	Scan/Cobol	A
IBM Main DOS,MVS OS and VM	Cobol	Superstructure	A
IBM Main OS	Cobol	SYDOC	A
IBM Main MVS	Cobol	Via/Insight (PWB)	A
IBM Main OS	any partitioned d.s.	VSearch	A

Table 1

Code Analyzer Tools

Cross Reference Tools

Environment	Language	Product Name	App
IBM OS	Cobol	AUTOREF	A
IBM Main MVS, OS DOS and VM	Cobol	Byblos - Source Documentation Sys.	A
IBM Main MVS,DOS VM and VS1	None Spec	CA-Optimizer	A
IBM Main DOS, OS	Any	CICS-OFLU	A
Any w/ANSI Cobol	Cobol	COBXREF	A
Main	Cobol	CoPack	A
IBM Main DOS, OS	Cobol	Crossmacs	A
IBM Main OS,MVS, and DOS	Cobol	DCD II (PWB)	A
IBM Main DOS	Any	Dossier Browse	A
IBM Main DOS	None Spec	Dossier Prove	A
IBM Main OS,MVS	None Spec	Facs	A
Main, Mini	Cobol	FLOBOL	A
IBM Main DOS, OS	None Spec	Help	A
IBM Main MVS, OS VM and VS1	None Spec	Illustrate	A
IBM Main MVS, OS VM and VS1	JCL	JCL Check	A
IBM Main DOS, OS	JCL	JCL Flow Documentation System	A
IBM Main DOS, OS	JCL	JCL Xref	A
IBM Main MVS, DOSwVSE, OS	Cobol	LogiChain	A

Table continued on next page

Table 2

Cross Reference Tools

Cross Reference Tools (continued)

Environment	Language	Product Name	App
None Specified	Cobol	MAP	B
IBM Main OS	any partitioned data set	OSXref	A
IBM Main MVS,	Cobol	PM/SS	A
None Specified	Cobol/JCL	PSA/PSL	B
IBM Main DOS,MVS	Any/w/IDMS	Quality Assurance Tool Kit	A
IBM Main DOS, OS	None Spec	Quikjob	A
IBM Main	Cobol	SAGE Maintenance Programming System	A
IBM Main DOS,MVS OS and VM	Cobol	Scan/Cobol	A
IBM Main DOS,MVS	Cobol	SMU Series	B
IBM Main VM, MVS	Cobol	SofTool Programming Environment	A
IBM Main MVS, OS VM and VS1	None Spec	Software Cross Check	A
IBM Main OS	Cobol	SYDOC	A
IBM Main MVS	Cobol	Via/Insight (PWB)	A
IBM Main OS	any partitioned data set	VSearch	A
IBM Main OS	JCL	VXRef	A

Table 2

Cross Reference Tools (continued)

Documentation Aid Tools

Environment	Language	Product Name	App
Custom Fitted	None Spec	ACT	B
IBM Main MVS	Cobol	Automatic Documentation Facility	A
None Specified	High Level	Battlemap	B
IBM Main MVS, OS DOS and VM	Cobol	Byblos - Source Documentation Sys.	A
IBM Main OS, DOS	Cobol	CA-Optimizer	A
IBM Main OS	Cobol	Cobol - Warnier Generator	A
Main	Cobol	CoPack	A
IBM Main	Cobol	DataTEC	B
IBM Main OS,MVS, and DOS	Cobol	DCD II (PWB)	A
IBM Main MVS, OSwTSO	Cobol	Diagraphics For Data Processing	A
Any w ANSI Cobol	Cobol	Doc-F	A
IBM Main DOS	Any	Dossier Browse	A
IBM Main DOS	None Spec	Dossier Prove	A
IBM Main MVSwTSO VMwCMS	Cobol	Fastbol	A
Main, Mini	Cobol	FLOBOL	A
IBM Main DOS, OS	None Spec	Help	A
IBM Main VM, VMS	Cobol	Interface Documentor	A
IBM Main MVS, OS VM and VS1	JCL	JCL Check	A

Table continued on next page

Table 3

Documentation Aid Tools

Documentation Aid Tools (continued)

Environment	Language	Product Name	App
IBM Main DOS, OS	JCL	JCL Flow Documentation System	A
IBM Main DOS, OS	JCL	JCL Xref	A
IBM Main MVS, DOSwVSE, OS	Cobol	LogiChain	A
IBM Main OS	any partitioned data set	OSXref	A
IBM Main OS. MVS	Cobol	Pathvu	A
IBM Main MVS, OSwTSO	Cobol	PM/SS	A
None Specified	Cobol/JCL	PSA/PSL	B
IBM Main DOS, OS	None Spec	Quikjob	A
IBM Main OS	Cobol	RECFLOW	A
IBM Main MVS, VM	Cobol	Recoder	A
IBM Main DOS, VM	Any	Res-Q	A
None Specified	Cobol	Reverse Engineering	A
IBM Main DOS,MVS OS and VM	Cobol	Scan/Cobol	A
IBM Main DOS,MVS	Cobol	SMU Series	B
IBM Main VM, MVS	Cobol	SofTool Programming Environment	A
IBM Main DOS, OS	Cobol	Source Program and JCL Documentor	A
IBM Main DOS,MVS OS and VM	Cobol	Superstructure	A
IBM Main OS	Cobol	SYDOC	A

Table continued on next page

Table 3

Documentation Aid Tools (continued)

Documentation Aid Tools (continued)			
Environment	Language	Product Name	App
IBM Main OS	any partitioned data set	VSearch	A
IBM Main OS	JCL	VXRef	A
IBM Main OS, DOS	Cobol	Wizard Compare	A
None Specified	Cobol	XPF	B

Table 3

Documentation Aid Tools (continued)

Execution Monitor/Debug Tools			
Environment	Language	Product Name	App
IBM Main OS, MVS	Cobol	Analyzer (PWB)	B
IBM Main DOS, OS MVS	Cobol	CICS Interactive Cobol Debug System	A
IBM 360,370 wDOS OS	Cobol	JSA DEBUG - Cobol Debug	A
IBM Main OS	Cobol	Quick Online Debugging System	A
IBM 360,370,30XX 43XX	Any	Trace	A
IBM Main MVS/XA	Cobol	XPF/Cobol	A

Table 4

Execution Monitor/Debug Tools

Data Manipulation Tools

Environment	Language	Product Name	App
IBM Main DOS, OS	Any	CICS - OLFU	A
IBM Main	Cobol	DataTEC	B
IBM Main MVSwTSO	Cobol	Data- Xpert (PWB)	B
IBM TSOwISFP, IMS/DC, MVS, MVS/XA	Cobol	IMS - Expert	B
IBM Main DOS, OS	Any	Matchmaster	A
IBM Main MVS, OSwTSO	Cobol	PM/SS	A
IBM Main DOS,MVS VS1wIDMS/R	Any w IDMS	Quality Assurance Tool Kit	B
IBM Main DOS, OS	None Spec	Quikjob	A
IBM Main DOS, OS	Cobol	ReadCobol	A
None Specified	Cobol	Reverse Engineering	A
IBM Main DOS, VM	Any	Res-Q	A
Any w ANSI Cobol	Cobol	SCobol	A
IBM Main DOS,MVS	Cobol	SMU Series	B
IBM Main MVS, OS VM, and VS1	None Spec	Software Cross Check	A
IBM Main OS	Any Partitioned d. s.	VSearch	A

Table 5

Data Manipulation Tools

Data Standardization Tools

Environment	Language	Product Name	App
IBM Main OS	Cobol	Cobol Structuring Aid (PWB)	A
IBM Main	Cobol	DataTEC	B
IBM Main DOS	None Spec	Dossier Prove	A
IBM Main OS, DOS	Cobol	Hawkeye (PWB)	A
IBM Main MVS, OS VM and VS1	JCL	JCL Check	A
IBM Main DOS, OS	Any	Matchmaster	A
IBM Main MVS, OSwTSO	Cobol	PM/SS	A
IBM Main DOS,MVS VS1wIDMS/R	Any w IDMS	Quality Assurance Tool Kit	B
IBM Main DOS, OS	None Spec	Quikjob	A
IBM Main DOS, OS	Cobol	ReadCobol	A
IBM Main DOS,MVS	Cobol	SMU Series	B
IBM Main VM, MVS	Cobol	SofTool Programming Environment	A
IBM Main MVS, OS VM, and VS1	None Spec	Software Cross Check	A
IBM Main DOS, OS	Cobol	Source Program Compare	A
None Specified	None Spec	TDGEN	B
IBM Main OS	Any	Transfixxer	B
IBM Main OS	Any Partitioned d. s.	VSearch	A
IBM Main OS	JCL	VXRef	A

Table 6

Data Standardization Tools

File Comparator Tools

Environment	Language	Product Name	App
IBM Main OS, DOS	Cobol	Comparex (PWB)	B
Any w ANSI Cobol	Cobol	Diffs	A
IBM Main MVS, OS VM w CMS	Cobol	Librarian	A
IBM Main DOS, OS	None Spec	Quikjob	A
IBM Main DOS, VM	Any	Res-Q	A
IBM Main	Cobol	SAGE Maintenance Programming System	A
IBM Main DOS,MVS	Cobol	SMU Series	B
IBM Main OS	Any Partitioned d. s.	VSearch	A
IBM Main OS	JCL	VXRef	A
IBM Main OS, DOS	Cobol	Wizard Compare	A

Table 7
File Comparator Tools

Program Management/Change Control Tools

Environment	Language	Product Name	App
IBM Main VSE, VM and MVS	None Spec	CA-Unicenter	B
IBM Main	None Spec	Change and Configuration Control	B
IBM Main MVS	None Spec	Change-Man	B
IBM Main	None Spec	Endevor	B
None Specified	None Spec	Smarts	B
None Specified	None Spec	S-TCAT	B

Table 8

Program Management/Change Control Tools

Reformatter Tools

Environment	Language	Product Name	App
IBM Main	Cobol	ASTEC	B
IBM Main DOS	None Spec	CA-Converter	A
IBM 360/70 w OS	Cobol	Cobol Recomposition System	A
IBM Main OS	Cobol	Cobol Structuring Aid	A
IBM Main	Cobol	CoPack	A
IBM Main OS	Cobol	Enforce	A
IBM Main OS, DOS	Cobol	Hawkeye (PWB)	A
IBM Main DOS, OS	Any	Matchmaster	A
IBM Main DOS, OS	None Spec	Quikjob	A
IBM Main DOS, VM	Any	Res-Q	A
IBM Main MVS, OSwTSO	Cobol	PM/SS	A
IBM Main DOS,MVS VS1wIDMS/R	Any w IDMS	Quality Assurance Tool Kit	B
IBM Main DOS, OS	None Spec	Quikjob	A
IBM Main DOS, OS	Cobol	ReadCobol	A
IBM Main MVS, VM	Cobol	Recoder	A
IBM Main DOS, OS	Cobol	Reformat	A
IBM Main DOS, VM	Any	Res-Q	A
IBM Main	Cobol	SAGE Maintenance Programming System	A
Any w ANSI Cobol	Cobol	SCobol	A
IBM Main DOS,MVS	Cobol	SMU Series	I

Table continued on next page

Table 9

Reformatter Tools

Reformatter Tools (continued)

Environment	Language	Product Name	App
IBM Main DOS,MVS OS and VM	Cobol	Superstructure	A
IBM Main OS	Any Parti-tioned d. s.	VSearch	A

Table 9

Reformatter Tools (continued)

Restructurer Tools

Environment	Language	Product Name	App
None Specified	High Level	Battlemap	B
IBM Main MVS,DOS and OS	VS Cobol II	COBOL Structuring Facility	A
IBM Main MVS, OSwTSO	Cobol	PM/SS	A
IBM Main MVS, VM	Cobol	Recoder	A
IBM Main DOS, OS	Cobol	Reformat	A
IBM Main DOS, VM	Any	Res-Q	A
IBM Main	Cobol	RETROFIT	B
IBM Main OS	Cobol	Structured Retrofit	A
IBM Main DOS,MVS OS and VM	Cobol	Superstructure	A

Table 10

Restructurer Tools

Source Code Comparator Tools

Environment	Language	Product Name	App
IBM Main OS, DOS	Cobol	Comparex (PWB)	B
Any w ANSI Cobol	Cobol	Diffs	A
IBM Main MVS, OS VM w CMS	Cobol	Librarian	A
IBM Main MVS, DOSwVSE, OS	Cobol	LogiChain	A
None Specified	Cobol	MAP	B
IBM Main DOS, OS	None Spec	Quikjob	A
IBM Main DOS, VM	Any	Res-Q	A
IBM Main	Cobol	SAGE Maintenance Programming System	A
IBM Main MVS, OS	Cobol	S/Compare	A
IBM Main DOS,MVS	Cobol	SMU Series	B
IBM Main DOS, OS	Cobol	Source Program Compare	A
IBM Main OS	Cobol	Text Comparator	A
IBM Main OS, MVS VM	Cobol	Trailblazer	A
IBM Main OS	Any Partitioned d. s.	VSearch	A
IBM Main OS, DOS	Cobol	Wizard Compare	A

Table 11

Source Code Comparator Tools

Test Case Monitor Tools

Environment	Language	Product Name	App
Custom Fitted	None Spec	ACT	B
IBM Main OS, DOS	Cobol	Advanced Debugging System	A
IBM Main OS, MVS	Cobol	Analyzer (PWB)	B
IBM Main OS, DOS	Cobol	CA-Optimizer	A
IBM Main OS, VM, MVS and VS1	None Spec	Final Test	A
IBM Main DOS,MVS	Cobol	SMU Series	B
IBM Main VM, MVS	Cobol	SofTool Programming Environment	A
None Specified	None Spec	TCAT	A
IBM Main VM, MVS	Cobol	Testing Instrumenters	A
IBM 370, 30XX, 43XX	Cobol	Trace	A
IBM Main OS, MVS VM	Cobol	Trailblazer	A
IBM MVS, MVS/XA, VM/CMS, TSO	Cobol	XPEDITOR	A
None Specified	Cobol	XPF	B
IBM Main MVS/XA	Cobol	XPF/Cobol	A

Table 12

Test Case Monitor Tools

Translator Tools			
Environment	Language	Product Name	App
IBM Main DOS, OS	Any	Matchmaster	A
IBM Main	ASM to COB	ReAct	B
Any w ANSI Cobol	Cobol	SCobol	A
IBM Main MVS/TSO	Cobol	Transit	B

Table 13

Translator Tools

APPENDIX A

A Survey of Software Maintenance Tools That Enhance Program Understanding

by H.B. Holbrook and S.M. Thebaut

SERC-TR-9-F

Software Engineering Research Center
University of Florida
Gainesville, Florida 32611

TABLE OF CONTENTS (SERC Report)

TABLE OF CONTENTS (SERC Report, continued)

TABLES

A SURVEY OF SOFTWARE MAINTENANCE TOOLS THAT ENHANCE PROGRAM UNDERSTANDING

H.B. Holbrook and S.M. Thebaut

Computer and Information Sciences Department
University of Florida
Gainesville, FL 32611

ABSTRACT

This report summarizes the results of a recent survey of commercially available software tools which purport to aid in the task of program understanding. The effort was undertaken in connection with the SERC "Maintenance Assistant" Research Project at the University of Florida during the summer of 1987, and resulted in the identification of 116 tools. Most of the tools identified provide insight into the program structures and operations considered important for program comprehension.

1. INTRODUCTION

1.1. Background

In May of this year, the (SERC) "Maintenance Assistant" project was commissioned to study the problems of software maintenance and to investigate the concept of bringing together a combination of loosely integrated tools that could improve the productivity of maintenance programmers and increase the reliability of modified programs. One area of study has been that of "reverse software engineering" - the process of recovering "lost" or otherwise unavailable program requirements or design specifications that can aid in understanding the program and modifying it. Background work for study in this area included an examination of existing software tools which may aid in the task of program understanding. This survey is a result of that effort.

1.2. The Problem

"Performing maintenance, first and foremost, requires an understanding of the program - its functional objective, its internal structure, and its operational requirements." With that, Martin and McClure [1] summarize the relationship between software maintenance and program understanding. Unfortunately, the situation described by Fay and Holmes [2] below is familiar to many.

> *You've been barraged with all of the facts: the original programmers are gone; the few comments that are in the code aren't necessarily correct (although they might be); and the small amount of documentation that exists (if any) is not necessarily correct or complete - it hasn't been updated for the last who-knows-how-many code updates.*

Traditionally, programmers have had little automated support in dealing with such situations. Programs have been "analyzed" by trying to mimic computer functions with the aid of listings, paper clips, highlighters, and the maintainer's own limited mental capacity for dealing with masses of repetitive data.

There now exist, however, tools that can provide users with relevant information on various aspects of a program (or system) in convenient forms. This report provides an overview of these tools.

1.3. Understanding Programs

As suggested by Brooks [3] and others, understanding a program is a process which involves reconstructing mappings from the programming domain to the problem domain. A programmer accomplishes this by generating a hierarchy of hypotheses about how a program meets a user's needs. From this perspective, *understanding* occurs when a programmer verifies (or adjusts) his hypotheses based on structures or operations found within the code. The classes of tools we have identified here support what Brooks calls "the verification process" of finding the operations and structures within a program that satisfy the hypotheses generated.

Another view of program understanding is offered by Letovsky and Soloway [4], and involves what are called "programming plans." These correspond roughly to Brooks' hypotheses. The understanding process here is associated with recognizing the plans in the existing code. Unfortunately, plans can be hard to recognize in the code if they are "delocalized," or spread through code that performs other functions. Many of the surveyed tools can be very useful in isolating delocalized plans.

1.4. Scope of Effort

Our research in reverse engineering is based, in part, on the premise that the only reasonably certain things a maintenance programmer will have to work with are the latest version of the source program in question and the system (hardware and operating system) on which it runs. Consequently, we have not investigated tools which require the availability of "external" information such as design specifications or other documentation, or which require adherence to a particular development methodology.

Furthermore, while we have tried to be thorough, the survey is by no means exhaustive and the tools we have listed are meant to be representative of the techniques under study.

2. METHODOLOGY

2.1. Sources

Tool information was obtained from several sources in our survey. One was product descriptions provided to us by vendors advertising in national software-oriented periodicals such as *Software Maintenance News* (SMN), an independent monthly which publishes articles dealing with all aspects of software maintenance [5].

Another source was a 1986 article by David Roman appearing in *Computer Decisions* which provided, in tabular form, a list of some 85 software maintenance tools along with their environments and functions [6]. The article also suggests a model for categorizing maintenance tools according to the functions provided.

A third source, the Office of Software Development and Information Technology of the US General Services Administration, publishes the *Software Aids and Tools Survey* which covers some 300 tools that apply to software engineering in general [7]. Information provided includes tool function, source language, cost, and producer.

Finally, the *ICP Software Directory* [8] consists of six volumes and some 18,000 software product and service descriptions for nearly all application areas. One volume is dedicated to systems software applications and covers many tools that deal with software maintenance. The directory provides source and environment information as well as a brief description of each tool.

2.2. Tool Categories

As we began our survey, it became obvious that we needed a reasonable way of classifying the functions of the tools identified. As a starting point, the US General Services Administration (GSA) categories as defined for their *Programmer's Workbench* project were utilized [6].

The *Programmer's Workbench* came about as the result of an effort to help federal information processing managers maintain their ever-growing mass of software. The categories identified functionally group tools the GSA felt were fundamental to maintenance programming. They are: test coverage monitors, translators, code reformatters, data standardization tools, cross referencers, documentation aids, file comparators, data manipulation tools, restructurers, and code analyzers.

We considered seven of the ten GSA categories to be appropriate for our purposes, and added another. The categories we used are: test coverage monitors, code reformatters, cross referencers, documentation aids, restructurers source comparators, code analyzers, and execution monitors/debuggers. Descriptions of these categories are given in Section 4.

2.3. Storing the Data

Data from the survey have been stored in a Unify database and are available to SERC affiliates in various formats upon request. Tools are represented as separate records with the following fields (length in characters in parentheses): name (40), company (35), phone (15), function (25), environment (50), language (40), and source (12).

3. CATEGORIES AND REPRESENTATIVE TOOLS

In this section, the various categories of tools that aid in understanding computer programs are described. The basis for these categories is the GSA *Programmer's Workbench* project with modifications as explained in the previous section.

For each category, a brief description of the associated functions is provided. Examples of tools providing the specific functions described are given in parentheses. Following the category description, a summary list of representative tools is provided.

3.1. Code Analyzers

This group includes tools which statically analyze a program's control structure and data flow. By *statically*, we mean that the program itself is not executed and therefore its run-time behavior is not examined.

Code analysis tools vary greatly in the functions they perform. Most, however, fit within one of two categories: batch-oriented metrics generators, and interactive logic browsers.

The batch-oriented metrics generators produce measures which may be compared to predefined standards in order to assess the complexity of the software. In some cases, these tools may be used to determine the applicability of restructuring techniques (Pathvu, Superstructure).

Some of the interactive tools have been specifically designed to facilitate program understanding (Fastbol, Via/Insight). They provide the means for navigating through a program's logic or data flow by isolating specified classes of source statements (such as input, conditions, or particular control structures).

Other functions provided by code analyzers include the highlighting of "dead code" and the identification of other questionable coding practices (Basic Program Analyzer).

Table 1 table lists tools that perform code analysis.

3.2. Documentation Aids

These tools generate graphical documentation which illustrates program logic at various levels of abstraction. Examples of output are flowcharts (Flowgen/F II, Flobol), calling hierarchies (Pathvu, Tree Diagrammer), and Warnier-Orr diagrams (Cobol-Warnier Generator).

Table 2 lists tools that generate program documentation.

3.3. Cross Referencers

These tools trace the use of data elements, named paragraphs, or procedures through a program. This is important to program understanding in that a function is very often recognized as a characteristic sequence of operations on a data structure. Thus, cross referencers may help in isolating

Table 1

Code Analysis Tools

Tool Name	Language Supported
Basic Program Analyzer	Basic
C-Tracer	C
Cobol Structuring Facility	VS Cobol II
F-Scan	Fortran
Fastbol	Cobol
Fortran Static Code Analyzer	Fortran
ISAS (Integrated Software Analysis)	Fortran,Assembler
LOGISCOPE	Pascal,C,Fortran,Cobol,Modula2
LogiChain	Cobol
PM/SS	Cobol
Pathvu	Cobol
RSVP80 Series	Fortran
Reftran	Fortran
Reverse Engineering	Cobol,Fortran
SAGE Maintenance Programming System	Cobol
Sauer XRef	Basic
Scan/Cobol	Cobol
Superstructure	Cobol
Sydoc	Cobol
VAX Source Code Analyzer	Multiple languages
Via/Insight	Cobol
Wiseman	Basic

what Letovsky and Soloway call "delocalized plans" in which a particular function is spread throughout sections of seemingly unrelated code [4].

Object references are usually identified by source statement numbers. Associated with the statement numbers may be additional information such as the type of statement involved (move, assignment, conditional, etc.) or perhaps a copy of the statement itself.

Tool output is typically either a printed report or an on-line display. DCD II, for example, provides general data usage information within a Cobol data definition section of a source code listing. Optionally, it will produce an expanded cross reference listing that summarizes the types of operations associated with each data item.

Other tools operate from within an editor (Fastbol, Via/Insight) and provide on-line access to the information. Fastbol, for example, provides a cross-reference chart which displays how a data item is derived. From this display, the user may then go directly to the associated source code with a keystroke.

Some cross-reference tools allow the user to trace redefined data items (DCD II, Fastbol). This is especially important in dealing with Cobol group definitions or aliases. Some of the general file search tools (Dossier Browse, CICS/OLFU) may not provide this feature since they are language independent.

Table 3 lists tools that perform cross referencing.

3.4. Restructurers

Restructurers accept unstructured code as input and produce a structured program with the same functionality as output. The advertised advantage to structured code is that it enhances human readability and understanding by providing a hierarchical arrangement that allows for a quick grasp of the global as well as local structure of the program [9]. Moreover, the resultant code is of a consistent style - a situation which is not usual after numerous modifications by different programmers. There are, however, some questions about the usefulness of these tools. Interested readers may wish to review the October, November, and December 1986 issues of *Software Maintenance News* for

Table 2

Program Documentation Tools

Tool Name	Language Supported
ADF (Automatic Documentation Facility)	Cobol,Assembler
ADPL	Pascal,C,Fortran
ADS (Automatic Documentation System)	Cobol,Assembler
Autodoc II	Assembler
Basic Program Analyzer	Basic
Byblos-Source Documentation System	Cobol
CoPack	Cobol
Cobol Glossary	Cobol
Cobol-Warnier Generator	Cobol
DCD II	Cobol
DFDP (Diagraphics for Data Processing)	Cobol
Doc-F	Cobol
Docu/Manager	RPG
Documentation System (DOC)	Assembler,Basic
Dossier Browse	Any
Dossier Prove	None specified
FLOBOL	Cobol
Fastbol	Cobol
Flowgen/F II	Fortran
Help	None specified
Interface Documentor	Fortran,Cobol,Assembler
JCL XRef	JCL
JCLCheck	JCL
JCLFlow Documentation System	JCL
LOGISCOPE	Pascal,C,Fortran,Cobol,Modula2
LogiChain	Cobol
MAD/3000	Cobol,Fortran,Basic
OS XRef	Any partitioned data set
PM/SS	Cobol
Pathfinder	RPG
Pathvu	Cobol
Quikjob	None specified
RPG Flowchart Utility	RPG
RSVP80 Series	Fortran
Rand Development Center SMU Series	Cobol
Recflow	Cobol
Recoder	Cobol
Reftran	Fortran
Res-Q	Any
Reverse Engineering	Cobol,Fortran
Scan/Cobol	Cobol
Softool Programming Environment Tools	Fortran,Cobol,C
Source Program and JCL Documentor	Cobol,Assembler
Superstructure	Cobol
Sydoc	Cobol
Sysd	Assembler
TAMU Automate Flowchart System	Fortran
Tree Diagrammer	C,Basic,Pascal,DBASE,Fortran,Modula2
VSearch	Any partitioned data set
VXRef	JCL
Wizard Compare	Cobol
Wsdoc	APL

more information [5].

Table 4 lists tools that restructure source code.

Table 3

Cross Referencing Tools

Tool Name	Language Supported
ADPL	Pascal,C,Fortran
AutoRef	Assembler,Cobol
Basic Program Analyzer	Basic
Byblos-Source Documentation System	Cobol
CA-Optimiser	None specified
CICS-OLFU	Any
CoPack	Cobol
Cobxref	Cobol
Crossmacs	Cobol
DCD II	Cobol
Docu/Manager	RPG
Dossier Browse	Any
Dossier Prove	None specified
FLOBOL	Cobol
Facs	None specified
Help	None specified
ISAS (Integrated Software Analysis)	Fortran,Assembler
Illustrate	None specified
JCL XRef	JCL
JCLCheck	JCL
JCLFlow Documentation System	JCL
LogiChain	Cobol
MAD/3000	Cobol,Fortran,Basic
OS XRef	Any partitioned data set
PM/SS	Cobol
Pathfinder	RPG
Quality Assurance Tool Kit	Any using IDMS
Quikjob	None specified
RSVP80 Series	Fortran
Rand Development Center SMU Series	Cobol
Reftran	Fortran
SAGE Maintenance Programming System	Cobol
Sauer XRef	Basic
Scan/Cobol	Cobol
Softool Programming Environment Tools	Fortran,Cobol,C
Software Cross Check	None specified
Source Print	C,Basic,Pascal,DBASE,Fortran,Modula2
Sydoc	Cobol
Toolbox	C
VAX Source Code Analyser	Any
VSearch	Any partitioned data set
VXRef	JCL
Via/Insight	Cobol
Wsdoc	APL

3.5. Reformatters

Reformatters are intelligent text editors which enhance program understanding by manipulating the pagination, spacing, and indentation of program source code. The use of a reformatters can result in a uniform coding style for the programs being maintained.

One of the important elements of reformating is accurately depicting the scope of control statements, especially where there is complex nesting. This is done by key word alignment (If, else, while, begin, end, etc.) or by drawing lines to indicate scope (Source Print).

Some tools (CSA, Hawkeye) provide very useful features for reformating Cobol code. These include provisions to enforce naming and level standards, to alphabetize working storage entries, and to sequence paragraph names.

Table 4

Restructuring Tools

Tool Name	Language Supported
Cobol Structuring Facility	VS Cobol II
PM/SS	Cobol
Recoder	Cobol
Structured Retrofit	Cobol
Superstructure	Cobol

Table 5 lists tools that reformat source code.

Table 5

Reformating Tools

Tool Name	Language Supported
Basic Program Analyzer	Basic
Basic-Doc	Basic
CSA (Cobol Structuring Aid)	Cobol
CoPack	Cobol
Cobol Recomposition System	Cobol
Enforce	Cobol
Hawkeye	Cobol
Matchmaster	Any
Neater2	PL1
PM/SS	Cobol
Quikjob	None specified
RSVP80 Series	Fortran
Rand Development Center SMU series	Cobol
ReadCobol	Cobol
Recoder	Cobol
Reformat	Cobol
Res-Q	Any
SAGE Maintenance Programming System	Cobol
SCobol	Cobol
Source Print	C,Basic,Pascal,DBASE,Fortran,Modula2
Superstructure	Cobol
Toolbox	C
VSearch	Any partitioned data set
Wiseman	Basic

3.6. Execution Monitors/Debuggers

This group of tools allows the programmer to interactively monitor and manipulate the process of a program as it executes. In so doing, the maintenance programmer can directly examine the behavior of a program and the effects of various inputs.

Within this category, two basic functions are provided: tracing (Analyze, C-Tracer) and breakpointing (XPF/Cobol, JSCDebug). Tracing presents a history of a program's execution by building a record of various program statements as they are executed. The types of statements recorded can vary from paragraph names to variable names with their values. Traces are useful in identifying program paths for given conditions.

Breakpointing allows the user to halt an executing program at specified points (breakpoints) and examine or modify its data. This allows the user to interactively examine the effects of selected code segments and to explore the consequences of varying data values.

Table 6 lists tools that monitor program execution.

Table 6

Execution Monitoring/Debugging Tools

Tool Name	Language Supported
Analyzer	Cobol
C-Tracer	C
CICS Interactive Cobol Debugging System	Cobol
FBUG/1000	Fortran
Interactive Debugging Monitor (IDM)	RPG
Intertest/CICS	Assembler
JSADebug-Assembler Debug	Assembler
JSCDebug-Cobol Debug	Cobol
QUODS (Quick Online Debugging System)	Cobol
Superbug	Assembler
Trace	Any
Tracer	Fortran Assembler
XDebug	Assembler
XPF/Assembler	Assembler
XPF/Cobol	Cobol

3.7. Test Coverage Monitors

Tools which monitor test case coverage keep track of which parts of a program are executed when a given set of test data is run. This involves executing an "instrumented" version of the program with the test data provided. Test monitors can enhance a programmer's understanding of a program by identifying the code segments associated with particular user-oriented functions.

Test coverage monitor reports can vary in form from static charts and tabular displays (Trailblazer) to on-line displays which are updated dynamically during symbolic debugging (XPF/Cobol).

Table 7 lists tools which monitor test case coverage.

Table 7

Test Case Coverage Tools

Tool Name	Language Supported
Advanced Debugging Syst.	none specified
Analyzer	Cobol
CA-Optimizer	None specified
CCA (Code Coverage Analyzer)	Fortran
FUS (Fortran Utility System)	Fortran
Final Test	None specified
LOGISCOPE	Pascal,C Fortran,Cobol,Modula2
RSVP80 Series	Fortran
Rand Development Center SMU Series	Cobol
Sleuth/3000	Cobol
Snoop for CICS	None specified
Softool Programming Environment Tools	Fortran,Cobol,C
TVVT (TPS Validation, Verification)	Fortran,Jovial
Testing Instrumenters	Fortran,Cobol,C
Trace	Any
Trailblazer	Cobol
XPF/Assembler	Assembler
XPF/Cobol	Cobol

3.8. Source Comparators

These tools are designed to help programmers quickly identify changes between program versions. This can be a significant aid in determining the rationale for previous undocumented maintenance.

Table 8 lists tools that compare source listings.

Table 8

Source Code Comparison Tools

Tool Name	Language Supported
Comparex	Cobol
Coopers & Lybrand Source Compare	Cobol
Diffs	Cobol
ISAS (Integrated Software Analysis)	Fortran,Assembler
Librarian	Cobol,PL1,Assembler
LogiChain	Cobol
Matchbook	Assembler
Quikjob	None specified
Rand Development Center SMU series	Cobol
Res-Q	Any
S/Compare	C,Cobol,DDS,PL1,RPG,TEXT
SAGE Maintenance Programming System	Cobol
Source Program Compare	Cobol
Text Comparator	Cobol,Assembler
Trailblazer	Cobol
VSearch	Any partitioned data set
Wizard Compare	Cobol

4. CHARACTERISTICS OF THE TOOLS SURVEYED

In this section we provide a statistical summary of the tools identified with respect to function categories, source languages supported, and operating environments. Note that some tools are associated with more than one function category. In particular, 51 of the tools surveyed are associated with a single category, 36 are associated with two, and 29 are associated with three or more.

4.1. Number of Tools by Function and Language Supported

Table 9 shows the distribution of tools surveyed according to function and language.
The abbreviations used are as follows:

Abbreviation	Function	Abbreviation	Language
SC	Source Code Comparators	CBL	Cobol
RF	Reformatters	FTN	Fortran
RS	Restructurers	CLG	C Language
CA	Code Analyzers	BSC	Basic
CR	Cross Referencers	PSC	Pascal
DA	Documentation Aids	NLS	No Language Specified
TM	Test Case Monitors	ANY	Any Language
DE	Execution Monitors/Debuggers	OTH	Other

4.2. Number of Tools by Function and Environment

Table 10 shows the distribution of tools surveyed according to function and operating environment.

Table 9

Number of Tools by Function and Language Supported

Function	CBL	FTN	CLG	ASB	PSC	BSC	PL1	MDL2	RPG	ANY	NLS	OTH	Total
RF	13	2	2	0	1	4	1	1	0	3	2	1	26
RS	5	0	0	0	0	0	0	0	0	0	0	0	5
DA	27	11	4	7	3	4	0	0	3	4	3	2	54
CA	12	7	2	1	1	3	0	1	0	2	0	0	23
SC	12	1	1	4	0	0	2	0	1	2	1	1	18
CR	16	7	4	2	2	0	0	1	2	6	7	2	46
TM	9	7	3	2	1	0	1	1	0	1	3	1	19
DE	5	2	1	6	0	0	0	0	1	1	0	0	15
Total	56	21	9	18	4	8	3	3	5	9	12	4	117

Table 10

Number of Tools by Function and Environment

Computer	DE	RF	RS	CR	CA	TM	SC	DA	Total
IBM	12	20	5	35	15	14	16	44	93
DEC	2	4	0	4	3	4	0	5	14
Hewlett-Packard	2	0	0	1	0	2	1	1	5
Data General	2	2	0	0	2	0	0	0	4
Burroughs	0	0	0	2	2	0	1	1	3
Eclipse	1	1	0	0	1	0	0	0	2
Nova	1	1	0	0	1	0	0	0	2
Prime	0	1	0	1	2	1	0	1	2
Sperry	0	0	0	0	1	0	0	1	2
Honeywell	0	0	0	1	1	0	1	2	2
Sun	0	0	0	0	0	1	0	1	2
Apollo	0	0	0	0	0	1	0	1	2

References

[1] Martin, James and McClure, Carma, *Software Maintenance: The Problem and its Solutions,* Prentice-Hall, Englewood Cliffs NJ, 1983.

[2] Fay, Sandra D., and Holmes, Denise G., "Help! I Have to Update an Undocumented Program," *Proceedings of the Conference on Software Maintenance,* IEEE Computer Society Press, 1985, pp. 194-202.

[3] Brooks, Ruven, "Towards a Theory of the Comprehension of Computer Programs," *International Journal of Man-Machine Studies,* Vol. 18, 1983, pp. 543-554.

[4] Letovsky, Stan and Soloway Elliot, "Strategies for Documenting Delocalized Plans," *Proceedings of the Conference on Software Maintenance,* IEEE Computer Society Press, 1985, pp. 144-151.

[5] Zvegintzov, Nicholas, (Ed.), *Software Maintenance News,* 56 Bay Street #400, Staten Island NY 10301.

[6] Roman, David; "Classifying Maintenance Tools," *Computer Decisions,* June 1986.

[7] "Software Aids and Tool Survey," *Office of Software Development and Information Technology,* U.S. Government Printing Office, 1986 GPO 022-002-00106-2.

[8] "ICP Software Directory," 55th Edition, International Computer Programs, Inc., 1986.

[9] DeBalbine, G., "Better Manpower Utilization Using Automatic Restructuring," *AFIPS Proceedings of the 1975 National Computer Conference,* AFIPS Press, 1975.

Appendix: Maintenance Tool Database

The following is an alphabetical listing of tools contained in the SERC Maintenance Tool Database. Function categories (based loosely on those of the GSA) are abbreviated as follows:

SC Source Code Comparitars - Identifies the differences between original and modified source listings.

FC File Comparitars - Compare data files, even if organized differently.

TR Translators - Convert programs from one language or version to another. Also may convert between operating systems.

RF Reformatters - Reformat source code to make it easier to read and maintain.

DS Data Standardization Tools - Locate and standardize data names.

RS Restructurers - Convert unstructured programs into structured ones to improve readability and maintainability.

CA Code Analyzers - Analyze the structure and logical flow of a program. Some also measure program complexity.

CR Cross Referencers - Trace the use of data elements, named procedures or code paragraphs through a program. Also note what other programs a data element appears in.

DA Documentation Aids - Automatically produce documentation for a program.

DM Data Manipulation Tools - Allow programmers to extract data from a file, modify it, and put it into file structures. Useful for constructing test data.

TM Test Case Monitors - Show which parts of a program are covered during execution.

DE Execution Monitors/Debuggers - Allow programmers to interactively monitor and interact with an executing program.

The source languages that a tool supports are abbreviated as follows:

CBL	Cobol
FTN	Fortran
CLG	C Language
BSC	Basic
PSC	Pascal

The sources for tool information are abbreviated as follows:

ICP *ICP Software Directory*, 55th Edition, 1986.

CMT "Classifying Maintenance Tools," *Computer Decisions*, June 1986.

FTS *Software Aids and Tool Survey*, Office of Software Development and Information Technology, 1986.

VSM Vendor Supplied Material

ADF (Automatic Documentation Facility)

Company: A+ Software
Phone: (315) 685-6918
Function: DA
Environment: IBM main/MVS
Language: CBL ASB
Source: ICP

ADPL

Company: Advanced Computer Concepts
Phone: (813) 923-1811
Function: DS CR DA
Environment: DEC VAX/VMS, OSwTSO
Language: PSC CLG FTN
Source: CMT ICP

ADS (Automatic Documentation System)

Company: A+ Software
Phone: (315) 685-6918
Function: DA
Environment: IBM main/MVS
Language: CBL ASB
Source: ICP

Advanced Debugging Syst.

Company: Interactive Solutions
Phone: (201) 488-3708
Function: TM
Environment: IBM main/DOS, OS
Language: CBL ASB PL1
Source: CMT ICP

Analyzer

Company: ALDON Computer Group
Phone: (415) 839-3535
Function: TM DE
Environment: IBM main/MVS, OS Hewlett-Packard 3000
Language: CBL
Source: VSM

AutoRef

Company: Siegel Software Services
Phone: (408) 429-6400
Function: CR
Environment: IBM, BURR main, OS
Language: ASB CBL
Source: ICP

Autodoc II

Company: Charles Downs
Phone: (602) 996-1027
Function: DA
Environment: IBM main/DOS, MVS, OS
Language: ASB
Source: ICP

Basic Program Analyzer

Company: Expert Systems
Phone: (409) 543-9222
Function: CA CR DA RF
Environment: IBM PC micro/MSDOS
Language: BSC
Source: VSM

Basic-Doc

Company: Applied Business Systems
Phone: (714) 759-0582
Function: RF
Environment: DATA GEN ECLIPSE, NOVA/RDOS
Language: BSC
Source: ICP

Byblos-Source Documentation System

Company: SGLIOS-PLS
Phone: France33-1-776.44 36
Function: DA CR
Environment: IBM main/DOS, OS, MVS, VM
Language: CBL
Source: ICP

C-Tracer

Company: IPT Corp
Phone: (415) 494-7500
Function: CA DE
Environment: Data General Eclipse, Nova, DEC VAX, Micro VAX
Language: CLG
Source: ICP

CA-Converter

Company: Computer Associates
Phone: (516) 227-3300

Function RF
Environment IBM main/DOS
Language none specified
Source CMT ICP

CA-Optimizer
Company Computer Associates
Phone (516) 227-3300
Function TM CR
Environment IBM main/DOS, MVS, VM, VSI
Language none specified
Source CMT

CCA (Code Coverage Analyzer)
Company HRB-Singer
Phone (814) 238-4311
Function TM
Environment VAX 11/7 wVMS
Language FTN
Source FTS

CICS Interactive Cobol Debugging System
Company Virtual Systems Software
Phone (212) 940-0068
Function DE
Environment IBM main/DOS, OS, MVS
Language CBL
Source ICP

CICS-OLFU
Company MacKinney Systems
Phone (417) 882-8012
Function CR DM
Environment IBM main/DOS, OS
Language any
Source CMT ICP

CSA (Cobol Structuring Aid)
Company Marble Computer
Phone (304) 267-2941
Function RF DS
Environment IBM main/OS
Language CBL
Source CMT VSM

CoPack
Company Generated Systems
Phone (312) 668-0506
Function: DA CR RF
Environment: main
Language CBL
Source ICP

Cobol Glossary
Company: MacKinney Systems
Phone (417) 882-8012
Function: DA
Environment IBM main/DOS, OS
Language CBL
Source CMT ICP

Cobol Recomposition System
Company: Laurel Computer Services
Phone: (616) 897-8522
Function RF
Environment IBM 360/70wOS
Language CBL
Source ICP

Cobol Structuring Facility
Company: International Business Machines
Phone ---
Function CA RS
Environment IBM main/DOS, MVS, OS
Language VS CBL II
Source VSM

Cobol-Warnier Generator
Company: Structured Methods Inc
Phone (212) 741-7720
Function: DA
Environment IBM main/OS
Language CBL
Source ICP

Cobxref
Company: Software Consulting Services
Phone (215) 837-8484
Function CR
Environment any running ANSI Cobol
Language: CBL
Source CMT ICP

Comparex

Company: Sterling Software Marketing
Phone: (916) 635-5535
Function: FC SC
Environment: IBM main/DOS, MVS, OS
Language: CBL
Source: CMT ICP

Coopers & Lybrand Source Compare

Company: Cooper & Lybrand
Phone: (212) 536-2000
Function: SC
Environment: IBM 360/70wOS
Language: CBL
Source: ICP

Crossmacs

Company: Management and Computer Services
Phone: (215) 648-0730
Function: CR
Environment: IBM main/DOS, OS
Language: CBL
Source: ICP

DCD II

Company: Marble Computer
Phone: (304) 267-2941
Function: CR DA
Environment: IBM main/DOS, MVS, OS
Language: CBL
Source: CMT VSM

DFDP (Diagraphics for Data Processing)

Company: ADPAC
Phone: (415) 974-6699
Function: DA
Environment: IBM main/MVS, CSwTSO
Language: CBL
Source: CMT ICP

Diffs

Company: Software Consulting Services
Phone: (215) 837-8484
Function: FC SC
Environment: any running ANSI Cobol
Language: CBL
Source: CMT ICP

Doc-F

Company: Software Consulting Services
Phone: (215) 837-8484
Function: DA
Environment: any running ANSI Cobol
Language: CBL
Source: CMT

Docu/Manager

Company: Application Development Services
Phone: (714) 261-2543
Function: DA CR
Environment: IBM SYS 34/36wSSP
Language: RPG
Source: ICP

Documentation System (DOC)

Company: Software Development Co
Phone: (713) 440-6029
Function: DA
Environment: IBM main, mini, micro w PICK
Language: ASB BSC
Source: ICP

Dossier Browse

Company: Computer Concepts
Phone: (503) 297-4741
Function: CR DA
Environment: IBM main/DOS
Language: any
Source: CMT ICP

Dossier Prove

Company: Computer Concepts
Phone: (503) 297-4741
Function: DS CR DA
Environment: IBM main/DOS
Language: none specified
Source: CMT

Enforce

Company: The Productivity Group
Phone: (212) 678-7165
Function: RF
Environment: IBM main/OS
Language: CBL
Source: ICP

F-Scan

Company: International Logic Corp

Phone: (415) 989-7223

Function: CA

Environment: IBM main, PRIME/OS, PRIMOS, VMS, AOS

Language: FTN

Source: ICP

FBUG/1000

Company: Corporate Computer Systems Inc

Phone: (201) 946-3800

Function: DE

Environment: HP 1000/RTE

Language: FTN

Source: ICP

FLOBOL

Company: COSMIC

Phone: (404) 542-3265

Function: CR DA

Environment: main, mini

Language: CBL

Source: ICP

FUS (Fortran Utility System)

Company: Digital Solutions

Phone: (201) 549-1700

Function: TM

Environment: IBM main/OS, VM SPERRY 1100

Language: FTN

Source: CMT ICP

Facs

Company: Comp Act Data Systems

Phone: (818) 992-4361

Function: CR

Environment: IBM main/OS, MVS

Language: none specified

Source: CMT

Fastbol

Company: The Analytic Sciences (TASC)

Phone: (617) 944-6850

Function: CA DA

Environment: IBM main/MVSwTSO, VMwCMS

Language: CBL

Source: CMT VSM

Final Test

Company: Triangle

Phone: (408) 554-8121

Function: TM

Environment: IBM main/MVS, OS, VM, VS1

Language: none specified

Source: CMT

Flowgen/F II

Company: California Computer Products

Phone: (714) 821-2011

Function: DA

Environment: IBM, HONEYWELL, SPERRY, DEC, CDC

Language: FTN

Source: ICP

Fortran Static Code Analyzer

Company: COSMIC

Phone: (404) 542-3265

Function: CA

Environment: DEC PDP 11/70 DEC VAX/VMS

Language: FTN

Source: FTS ICP

Hawkeye

Company: Blackhawk Data

Phone: (312) 236-8473

Function: RF DS

Environment: IBM main/OS, DOS DEC

Language: CBL

Source: CMT VSM ICP

Help

Company: On-Line Documentation

Phone: (201) 825-8466

Function: CR DA

Environment: IBM main/DOS, OS

Language: none specified

Source: CMT

ISAS (Integrated Software Analysis)

Company: Systems & Software Engineering Ope

Phone: (602) 721-0500

Function: CA CR SC

Environment: IBM main/DOS, MVS, OS
Language: FTN ASB
Source: VSM

Illustrate
Company: Triangle
Phone: (408) 554-8121
Function: CR
Environment: IBM main/MVS, OS, VM, VS1
Language: none specified .
Source: CMT

Interactive Debugging Monitor (IDM)
Company: Soltron Inc
Phone: (512) 346-9924
Function: DE
Environment: IBM SYSTEM 34, 36/SSP
Language: RPG
Source: ICP

Interface Documentor
Company: Softool
Phone: (805) 683-5777
Function: DA
Environment: IBM main/VM, MVS DEC VAX
SUN APOLLO
Language: FTN CBL ASB
Source: ICP

Intertest/CICS
Company: On-Line Software
Phone: (201) 592-0009
Function: DE
Environment: IBM main/MVSwCICS
Language: ASB
Source: ICP

JCL XRef
Company: MacKinney Systems
Phone: (417) 882-8012
Function: CR DA
Environment: IBM main/DOS
Language: JCL
Source: CMT

JCLCheck
Company: Triangle
Phone: (408) 554-8121
Function: DS CR DA
Environment: IBM main/MVS, OS, VM, VS1
Language: JCL
Source: CMT ICP

JCLFlow Documentation System
Company: Consumer Systems
Phone: (312) 495-8822
Function: CR DA
Environment: IBM main/DOS, OS
Language: JCL
Source: CMT ICP

JSADebug-Assembler Debug
Company: Computer Consultants and Software
Phone: (213) 784-6722
Function: DE
Environment: IBM 360/370 wDOS, OS
Language: ASB
Source: ICP

JSADebug-Cobol Debug
Company: Computer Consultants and Software
Phone: (213) 784-6722
Function: DE
Environment: IBM 360/370 wDOS, OS
Language: CBL
Source: ICP

LOGISCOPE
Company: Verilog
Phone: (703) 354-0371
Function: CA DA TM
Environment: micro mini mainframe/VMS CMS
MVS UNIX MULTICS NOS
Language: PSC CLG FTN CBL MDL2
Source: VSM

Librarian
Company: Applied Data Research
Phone: (201) 874-9000
Function: SC FC
Environment: IBM main/DOS, OS, VMwCMS
Language: CBL PL1 ASB
Source: CMT ICP

LogiChain

Company: Applications Programming
Phone: (609) 234-0099
Function: SC CA CR DA
Environment: IBM main/DOSwVSE, MVS, OS Burroughs Honeywell
Language: CBL
Source: CMT

Lookat

Company: EDP Management
Phone: (619) 462-5400
Function: FC
Environment: DURR\ 'GHS main/MCP
Language: any
Source: FTS

MAD/3000

Company: Related Computer Technology
Phone: (817) 379-5565
Function: DA CR
Environment: HP 3000 wMPE
Language: CBL FTN BSC
Source: ICP

Matchbook

Company: Westinghouse Management Systems
Phone: (412) 256-2900
Function: SC
Environment: IBM main/DOS, VSE
Language: ASB
Source: CMT ICP

Matchmaster

Company: Palace Computer Services
Phone: (212) 608-8045
Function: TR RF DS DM
Environment: IBM main/DOS, OS DEC VAX/VMS PDP11/RSX-11M
Language: any
Source: CMT ICP

Neater2

Company: KSU Research Foundation
Phone: (913) 532-6311
Function: RF
Environment: IBM main/DOS, CS
Language: PL1
Source: CMT

OS XRef

Company: MacKinney Systems
Phone: (417) 882-8012
Function: CR DA
Environment: IBM main/OS
Language: any partitioned data set
Source: CMT ICP

PM/SS

Company: ADPAC
Phone: (415) 974-6699
Function: RF DS RS CA CR DA DM
Environment: IBM main/MVS, OSwTSO
Language: CBL
Source: CMT ICP VSM

Pathfinder

Company: Hawkeye Information Systems
Phone: (818) 997-6894
Function: CR DA
Environment: IBM SYSTEM/3
Language: RPG
Source: ICP

Pathvu

Company: The Catalyst Group
Phone: (312) 938-5367
Function: CA DA
Environment: IBM main/OS, MVS Burroughs A series
Language: CBL
Source: CMT VSM ICP

QUODS (Quick Online Debugging System)

Company: ISL International
Phone: (212) 514-8230
Function: DE
Environment: IBM main/OS
Language: CBL
Source: ICP

Quality Assurance Tool Kit

Company: DBMS
Phone: (312) 961-5700

Function: DS CR DM
Environment: IBM main/DOS, MVS, VS1wCullinet IDMS/R
Language: any interfacing with IDMS
Source: CMT ICP

QuikJob
Company: Goal Systems International
Phone: (614) 888-1775
Function: SC FC RF DS CR DA DM
Environment: IBM main/DOS, OS
Language: none specified
Source: CMT ICP

RPG Flowchart Utility
Company: P&O Falco Inc
Phone: (318) 746-7441
Function: DA
Environment: IBM 43XX main/DOS
Language: RPG
Source: ICP

RSVP80 series
Company: General Research
Phone: (805) 964-7724
Function: TM RF CA CR DA
Environment: IBM main/DOS, OS DEC VAX/VMS prime minis
Language: FTN
Source: CMT ICP VSM

Rand Development Center SMU series
Company: Rand Information Systems
Phone: (415) 769-5800
Function: RF TM SC FC DS CR DA DM
Environment: IBM main/DOS, MVS
Language: CBL
Source: CMT ICP VSM

ReadCobol
Company: Foundation for Software Engineering
Phone: (602) 955-1148
Function: RF DS DM
Environment: IBM main/DOS, OS
Language: CBL
Source: CMT ICP

Recflow
Company: Thorne Data Inc
Phone: (404) 998-2708
Function: DA
Environment: IBM main/OS
Language: CBL
Source: ICP

Recoder
Company: Language Technology
Phone: (617) 741-1507
Function: RF RS DA
Environment: IBM main/MVS, VM
Language: CBL
Source: CMT

Reformat
Company: EDP Management
Phone: (619) 462-5400
Function: RF
Environment: IBM main/DOS, OS
Language: CBL
Source: CMT

Reftran
Company: William R. DeHaan
Phone: (805) 964-7724
Function: CA CR DA
Environment: any running Fortran
Language: FTN
Source: CMT ICP

Res-Q
Company: Quality Systems
Phone: (312) 266-6060
Function: SC FC RF DS DA
Environment: IBM main/DOS, VM, OS
Language: any
Source: CMT ICP

Reverse Engineering
Company: Meta Systems
Phone: (313) 663-6027
Function: CA DA DM
Environment: ---
Language: CBL FTN
Source: VSM

S/Compare

Company: ALDON Computer Group
Phone: (415) 839-3535
Function: SC
Environment: IBM main/MVS, OS HP 3000 IBM system 38
Language: CL CBL DDS PL1 RPG TEXT
Source: VSM

SAGE Maintenance Programming System

Company: SAGE Systems
Phone: (301) 652-8680
Function: CA RF CR SC FC
Environment: IBM main
Language: CBL
Source: ICP

SCobol

Company: Software Consulting Services
Phone: (215) 837-8484
Function: TR RF DS
Environment: any running ANSI Cobol
Language: CBL
Source: CMT ICP

Sauer XRef

Company: Sauer Computer Systems
Phone: (800) 325-9494
Function: CR CA
Environment: mini/OS65U
Language: BSC
Source: ICP

Scan/Cobol

Company: Group Operations
Phone: (202) 887-5420
Function: CA CR DA DM
Environment: IBM main/DOS, MVS, OS, VM
Language: CBL
Source: CMT ICP VSM

Sleuth/3000

Company: Tower Software Inc
Phone: (213) 545-7073
Function: TM
Environment: HP 3000w/MPE
Language: CBL
Source: ICP

Snoop for CICS

Company: Interactive Solutions
Phone: (201) 488-3708
Function: TM
Environment: IBM main/DOS, OS
Language: none specified
Source: CMT

Softool Programming *Environment* Tools

Company: Softool
Phone: (805) 683-5777
Function: TM CR DA DM
Environment: IBM main/VM, MVS DEC VAX
Language: FTN CBL CLG
Source: CMT ICP

Software Cross Check

Company: Triangle
Phone: (408) 554-8121
Function: DS CR DM
Environment: IBM main/MVS, OS, VM, VS1
Language: none specified
Source: CMT

Source Print

Company: Aldebaran Laboratories
Phone: (800) 257-5773
Function: RF CR
Environment: IBM micro/PCDOS&compats
Language: C BSC PSC DBASE FTN MODULA2
Source: VSM

Source Program Compare

Company: MacKinney Systems
Phone: (417) 882-8012
Function: SC DM
Environment: IBM main/DOS, OS
Language: CBL
Source: CMT ICP

Source Program and JCL Documentor

Company: Paul Newcum Applications
Phone: (401) 231-5650
Function: DA
Environment: IBM main/DOS, OS

Language: CBL ASB

Source: ICP

Structured Retrofit

Company: The Catalyst Group

Phone: (312) 938-5367

Function: RS

Environment: IBM main/OS, MVS Burroughs A series

Language: CBL

Source: CMT VSM ICP

Superbug

Company: Technology Consulting Corporation

Phone: (203) 674-8621

Function: DE

Environment: IBM main/VM

Language: ASB

Source: ICP

Superstructure

Company: Group Operations

Phone: (202) 887-5420

Function: RF RS CA DA

Environment: IBM main/DOS, MVS, OS, VM

Language: CBL

Source: CMT VSM

Sydoc

Company: Syncsort

Phone: (201) 930-9700

Function: DA CR CA

Environment: IBM main/OS

Language: CBL

Source: ICP

Sysd

Company: H&W Computer Systems International

Phone: (208) 377-0336

Function: DA DM

Environment: IBM main/MVS, OS

Language: ASB

Source: CMT ICP

TAMU Automate Flowchart System

Company: COSMIC

Phone: (404) 542-3265

Function: DA

Environment: IBM main

Language: FTN

Source: ICP

TVVT (TPS Validation, Verification)

Company: AMG Associates

Phone: (703) 892-5600

Function: TM

Environment: VAX 11/VMS DEC 20

Language: FTN JOVIAL

Source: FTS

Testing Instrumenters

Company: Softool

Phone: (805) 683-5777

Function: TM

Environment: IBM main/VM, MVS DEC VAX SUN APOLLO

Language: FTN CBL CLG

Source: ICP

Text Comparator

Company: Dataware

Phone: (716) 674-9310

Function: SC

Environment: IBM main/OS

Language: CBL ASB

Source: CMT ICP

Toolbox

Company: The Toolsmith

Phone: (916) 753-5040

Function: CR RF

Environment: DEC PDP11, VAX, CPM, VMS

Language: CLG

Source: ICP

Trace

Company: AK Inc

Phone: (408) 264-8015

Function: DE TM

Environment: IBM 370, 30XX, 43XX, PC/OS, MVS, PCDOS

Language: ANY

Source: ICP

Tracer

Company: IPT Corp
Phone: (415) 494-7500
Function: DE
Environment: DEC mini, DATA GENERAL mini
Language: FTN ASB
Source: ICP

Trailblazer

Company: The Analytic Sciences (TASC)
Phone: (617) 944-6850
Function: TM SC
Environment: IBM main/OS, MVS, VM
Language: CBL
Source: CMT VSM

Tree Diagrammer

Company: Aldebaran Laboratories
Phone: (800) 257-5773
Function: DA
Environment: IBM micro/PCDOS&compats
Language: CLG BSC PSC DBASE FTN MODULA2
Source: VSM

VAX Source Code Analyzer

Company: Digital Equipment Corporation
Phone: —
Function: CA CR
Environment: DEC main
Language: multi
Source: VSM

VSearch

Company: MB & Associates
Phone: (303) 794-1740
Function: CA SC FC RF DS CR DA DM
Environment: IBM main/OS
Language: any partitioned data set
Source: CMT ICP

VXRef

Company: MB & Associates
Phone: (303) 794-1740
Function: FC DS CR DA
Environment: IBM main/OS
Language: JCL
Source: CMT ICP

Via/Insight

Company: Viasoft
Phone: (602) 952-0050
Function: CR CA
Environment: IBM main/MVS
Language: CBL
Source: CMT VSM

Wiseman

Company: Qax International Systems
Phone: (904) 596-2090
Function: RF CA
Environment: Data General systems running Business Basic
Language: BSC
Source: ICP

Wizard Compare

Company: Wizard Computer Products
Phone: (803) 244-4110
Function: SC FC DA
Environment: IBM main/DOS, OS
Language: CBL
Source: CMT

Wsdoc

Company: IP Sharp Associates
Phone: (Can 416-364-5361)
Function: DA CR
Environment: IBM main/DOS, MVS
Language: APL
Source: ICP

XDebug

Company: Kolinar Corp
Phone: (408) 980-9411
Function: DE
Environment: IBM main/VM
Language: ASB
Source: ICP

XPF/Assembler

Company: Boole & Babbage
Phone: (408) 735-9550

Function: TM DE
Environment: IBM main/MVSwXA
Language: ASB
Source: CMT VSM

XPF/Cobol

Company: Boole & Babbage
Phone: (408) 735-9550
Function: DE TM
Environment: IBM main/MVSwXA
Language: CBL
Source: VSM ICP

APPENDIX B

Function Codes for Appendix B

Function Categories are abbreviated as follows:

CA Code Analyzer tool - Analyzes the structure and logical flow of a program. Some also measure program complexity.

CR Cross Reference tool - Traces the use of data elements, named procedures or code paragraphs through a program. Also notes appearance of data elements in other programs.

CG Code Generator tool - Generates source code from design specifications.

DA Documentation Aid tool - Automatically produces documentation for a program.

DE Execution Monitor/Debug tool - Allows programmers to interactively monitor and debug an executing program.

DM Data Manipulation tool - Allows programmers to extract data from a file, modify it, and re-insert it into file structures. Useful for constructing test data.

DS Data Standardization tool - Standardize data names between programs.

FC File Comparator tool - Compares data files even if they are organized differently.

MG Program Management/Change Control Tool - An aid to program management/change control

RF Reformatter tool - Reformats source code to make it easier to read.

RS Restructurer tool - Converts unstructured programs into structured ones to improve readability and maintainability.

SC Source Code Comparator tool - Identifies differences between original and modified source code.

TF Test File Generator tool - Prepares test data from information contained in source code.

TM Test Case Monitor tool - Shows which parts of a program are used during execution.

TR Translator tool - Converts programs from one language or version to another. Also may convert between operating systems.

Source Language Codes for Appendix B

The source languages that a tool supports are abbreviated as follows:

ASB	Assembler
BSC	Basic
CBL	Cobol
CLG	"C" Language
FTN	Fortran
PSC	Pascal
PL1	PL/1 Language
CICS	Customer Information Control System (IBM only)
RPG	Report Program Generator
MDL2	Modula2

Abend-AID for DB2
Company: Compuware Corp.
Phone: (800) 521-9353
Function: DE
Environment: IBM MVS/TSO/IMS
Language: DB2 (IBM mainframe DBMS)
Source: CW2

ACT
Company: McCabe Associates
Phone: (800) 638-6316
Function: CA DA TM
Environment: custom fitted
Language: none specified
Source: VSM

Analyzer (PWB)
Company: TravTech Inc.
Phone: (203) 277-9595
Function: TM DE
Environment: IBM MVS/OS, HP3000
Language: CBL
Source: VSM GSA

ASTEC
Company: MAINTEC, Inc.
Phone: (612) 831-2122
Function: RF
Environment: Mainframe only
Language: CBL
Source: VSM

AUDITEC
Company: MAINTEC, Inc.
Phone: (612) 831-2122
Function: RF RS
Environment: PC
Language: CBL
Source: VSM

Battlemap
Company: McCabe Associates
Phone: (800) 638-6316
Function: CA DA RS
Environment: none specified
Language: several high level
Source: VSM

CAPBAK
Company: Software Research, Inc.
Phone: (415) 957-1441
Function: TM
Environment: PC
Language: none
Source: VSM

CA-Unicenter
Company: Computer Associates
Phone: (800) 645-3003
Function: MG
Environment: IBM VSE/VM/MVS
Language: none
Source: VSM

Change and Configuration Control
Company: Softool Corporation
Phone: (805) 683-5777
Function: MG
Environment: DEC VAX, IBM, Honeywell,
DG, Sun, Gould
Language: none
Source: VSM

Change-Man
Company: SERENA Consulting
Phone: (800) 621-0851
Function: MG
Environment: IBM MVS
Language: none
Source: VSM

CICS Abend-AID
Company: Compuware Corp.
Phone: (800) 521-9353
Function: DE
Environment: IBM MVS/OS/TSO
Language: CICS
Source: VSM

CICS DBUG-AID
Company: Compuware Corp.
Phone: (800) 521-9353
Function: DE
Environment: IBM MVS, MVS/XA
Language: CICS
Source: VSM

CICS Playback
Company: Compuware Corp.
Phone: (800) 521-9353
Function: CA DE
Environment: IBM MVS/OS/TSO
Language: CICS
Source: VSM

COMPAREX (PWB)
Company: Sterling Software Mkting
Phone: (916) 635-5535
Function: FC SC
Environment: IBM DOS/MVS/OS
Language: CBL
Source: SRC GSA

DataTEC
Company: The Catalyst Group
Phone: (800) 323-3059
Function: DM DA DS
Environment: IBM,UNISYS,Honeywell
Language: CBL
Source: VSM

Data-Xpert (PWB)
Company: XA Systems Corporation
Phone: ---
Function: DM
Environment: IBM MVS/TSO
Language: CBL
Source: GSA

DBDS
Company: Sterling Software
Phone: (916) 635-5535
Function: DE
Environment: IBM MVS/DOS/VSE
Language: CICS
Source: VSM

DCD II (PWB)
Company: Marble Computer, Inc.
Phone: (800) 252-1400
Function: CR DA
Environment: IBM DOS/MVS/OS
Language: CBL
Source: SRC VSM GSA

Endevor
Company: Business Software Tech.
Phone: (617) 870-1900
Function: MG
Environment: IBM
Language: none specified
Source: VSM

Flowtec
Company: Maintec, Inc.
Phone: (612) 831-2122
Function: DA CA
Environment: PC
Language: none specified
Source: VSM

Foundation
Company: Authur Andersen and Co.
Phone: —
Function: CA DM CG
Environment: IBM mainframes
Language: DB2
Source: GC1

Hawkeye (PWB)
Company: Blackhawk Data Corp.
Phone: (312) 236-8473
Function: RF DS
Environment: IBM OS/DOS, DEC
Language: CBL
Source: SRC VSM GSA

IMS-Expert
Company: XA Systems Corp.
Phone: (800) 621-0854
Function: DM
Environment: IBM TSO/ISPF,IMS/DC
CICS,MVS,MVS/XA
Language: CBL PL1
Source: VSM

Integrated Software Analysis
Company: Systems & Software
Phone: (602) 721-0500
Function: CA CR SC
Environment: IBM DOS/MVS/OS
Language: FTN ASB
Source: VSM

Inspector
Company: Language Technology
Phone: (800) 732-6337
Function: CA
Environment: IBM OS/VS
Language: CBL
Source: VSM

LogiScope
Company: Verilog
Phone: (703) 354-0371
Function: CA DA TM
Environment: PC,UNIX,MULTICS,NOS
Language: PSC CLG FTN CBL MDL2
Source: VSM

MAP
Company: Amdahl Corporation
Phone: --
Function: CA CR SC
Environment: none specified
Language: CBL
Source: TS1

Maintenance Analysis Tool
Company: Science Applications, Inc.
Phone: --
Function: CA
Environment: none specified
Language: FTN
Source: TS2

PacBase
Company: CGI Systems, Inc.
Phone: --
Function: CA
Environment: Honeywell DPS8000
Language: CBL
Source: CSN

PATHVU (PWB)
Company: The Catalyst Group
Phone: (800) 323-3059
Function: CA DA
Environment: IBM,UNISYS,Honeywell
Wang, PCs
Language: CBL
Source: SRC VSM GSA

PolyDoc
Company: Polytron Corp.
Phone: (800) 547-4000
Function: DA CR
Environment: PC MS-DOS
Language: any
Source: VSM

PSA/PSL
Company: Meta Systems
Phone: (313) 663-6027
Function: DA CA CR
Environment: none specified
Language: CBL JCL FTN Others
Source: VSM

PVCS
Company: Polytron Corp.
Phone: (800) 547-4000
Function: MG
Environment: PC MS-DOS,VAX,mVAX
Language: any
Source: VSM

ReAct
Company: The Catalyst Group
Phone: (800) 323-3059
Function: TR
Environment: IBM,UNISYS,Honeywell
Language: ASM to CBL
Source: VSM

RETROFIT (PWB)
Company: The Catalyst Group
Phone: (800) 323-3059
Function: CA RS
Environment: IBM,UNISYS,Honeywell
Wang,PCs
Language: CBL
Source: VSM GSA

Roscoe DB2 Interface
Company: Applied Data Research
Phone: (201) 874-9000
Function: DE
Environment: IBM MVS,Roscoe
Language: Roscoe
Source: CW2

Smarts
Company: Software research Inc.
Phone: (415) 957-1441
Function: MG
Environment: none specified
Language: none specified
Source: VSM

SMU Series
Company: Rand Information Systems
Phone: (415) 769-5800
Function: RF TM SC FC DS CR DA DM
Environment: IBM DOS/MVS
Language: CBL
Source: VSM

S-TCAT
Company: Software research Inc.
Phone: (415) 957-1441
Function: MG
Environment: none specified
Language: none specified
Source: VSM

TCAT
Company: Software research Inc.
Phone: (415) 957-1441
Function: TM
Environment: none specified
Language: none specified
Source: VSM

TDGEN
Company: Software research Inc.
Phone: (415) 957-1441
Function: DM
Environment: none specified
Language: none specified
Source: VSM

Transfixxer
Company: Marble Computer
Phone: (800) 252-1400
Function: DM
Environment: IBM/OS
Language: any
Source: VSM

Transit (PWB)
Company: UCCEL Corp.
Phone: ---
Function: TR
Environment: IBM MVS/TSO
Language: CBL
Source: GSA

Traps
Company: TRAVTECH Inc.
Phone: (203) 277-9595
Function: TM DM
Environment: IBM TSO
Language: CICS IMS DB2
Source: VSM

Via/Insight (PWB)
Company: Viasoft
Phone: (602) 952-0050
Function: CR CA
Environment: IBM MVS
Language: CBL
Source: SRC VSM GSA

XPEDITOR
Company: Applic. Sys. Dev. Inc.
Phone: (800) 358-3048
Function: TM
Environment: IBM MVS,MVS/XA,VM/CMS,
TSO,ISPF,CICS,IMS/DC
Language: CBL, Roscoe, Hogan
Source: VSM

XPF
Company: Pansophic
Phone: (312) 954-2822
Function: TM DA
Environment: none specified
Language: CBL
Source: VSM

REFERENCES and SOURCES for Appendix B

SRC Holbrook, H. B. and Thebaut, S. M. A Survey of Software Maintenance Tools That Enhance Program Understanding. Gainesville, Fla.: Software Engineering Research Center, University of Florida, 1987.

VSM Vendor Supplied Materials and/or Advertisements.

GSA General Services Administration Programmer Workbench Project brochure (undated) Improving Productivity in Software Management.

FC1 "Tools Help DOE Center Organize Source Code," Federal Computer Week , Vol.2 No. 25, (June 20, 1988),46.

FC2 "Herdt Forges Codes of Excellence," Federal Computer Week , Vol 2 No. 30 (July 25, 1988), 36.

SMN "More Survey Responses, and Something Special in the Air," Software Maintenance News, Vol. 6 No. 6 (June 1988), 28.

SWC "Pick, 4GLs Are Tools That Can Unplug Applications Bottleneck," Software Canada, Vol. 5 (July-Aug, 1986), 8.

CW1 "Development Overload Creates Tool Pollution," Computerworld, Vol. 21 No. 46 (November 9,1987), 12.

CSN "Honeywell Adds to Its DPS8000," Computer Systems News, Vol. P10(1) (May 9, 1988), 22.

GC1 "Authur Andsersen Marketing Own CASE Software," Government Computer News, Vol. 7 No. 10 (May 13, 1988), 81.

TS1 Warren, S., "MAP: A Tool for Understanding Software," in Tutorial On Software Restructuring, ed. by Robert S. Arnold (Washington: IEEE Computer Society Press, 1986.), p. 153.

TS2 Berns, Gerald M., "Assessing Software Maintainability," in Tutorial On Software Restructuring, ed. by Robert S. Arnold (Washington: IEEE Computer Society Press, 1986.), p. 134.

CW2 "Competitors Learn to Play on Tilted Field," Computerworld, Vol. 22 No. 35 (August 29, 1988), 64.